

# On the Communication Complexity of Read-Once $AC^0$ Formulae

T. S. Jayram  
IBM Almaden

Email: jayram@almaden.ibm.com

Swastik Kopparty  
MIT

Email: swastik@mit.edu

Prasad Raghavendra  
University of Washington

Email: prasad@cs.washington.edu

**Abstract**—We study the 2-party randomized communication complexity of read-once  $AC^0$  formulae. For balanced AND-OR trees  $T$  with  $n$  inputs and depth  $d$ , we show that the communication complexity of the function  $f^T(x, y) = T(x \circ y)$  is  $\Omega(n/4^d)$  where  $(x \circ y)_i$  is defined so that the resulting tree also has alternating levels of AND and OR gates. For each bit of  $x, y$ , the operation  $\circ$  is either AND or OR depending on the gate in  $T$  to which it is an input. Using this, we show that for general AND-OR trees  $T$  with  $n$  inputs and depth  $d$ , the communication complexity of  $f^T(x, y)$  is  $n/2^{\Omega(d \log d)}$ . These results generalize the classical results on the communication complexity of set-disjointness [1], [2] (where  $T$  is an OR -gate) and recent results on the communication complexity of the TRIBES functions [3] (where  $T$  is a depth-2 read-once formula).

Our techniques build on and extend the information complexity methodology [4], [5], [3] for proving lower bounds on randomized communication complexity. Our analysis for trees of depth  $d$  proceeds in two steps: (1) reduction to measuring the information complexity of binary depth- $d$  trees, and (2) proving lower bounds on the information complexity of binary trees. In order to execute this program, we carefully construct input distributions under which both these steps can be carried out simultaneously. We believe the tools we develop will prove useful in further studies of information complexity in particular, and communication complexity in general.

**Keywords**—Communication complexity, Information complexity, AND-OR trees, Lower bounds

## I. INTRODUCTION

The communication complexity of functions belonging to  $AC^0$  has been an important area of study. Results in the two-party model have been pivotal for varied applications such as time-space tradeoffs, data structures decision trees (see [6] for references), and more recently for data streams [7], [5], [8]. Moreover, they have been instrumental in clarifying the individual and relative power of determinism, nondeterminism, and randomization. In the multiparty number-on-forehead model, non-trivial lower bounds for functions in  $AC^0$  have been shown recently [9], [10], [11], [12] that build on the pattern matrix method of Sherstov [13], [14]. The currently best known lower bound for a function in  $AC^0$ , is of the form  $\Omega(n^\varepsilon/2^{O(k)})$  for some constant

$\varepsilon < 1$  (the largest value of  $\varepsilon$  is plausibly  $3/10$ ) and the bound holds up to  $k = \Theta(\log n)$  players [12]. While the dependence on  $k$  is near-tight, extending these bounds to obtain a linear dependence on  $n$  for functions in  $AC^0$  is a significant challenge for the multiparty number-on-forehead model.

In this paper, we consider the two-party model and adopt the following convention for discussing different communication complexity measures of  $f$ :  $D(f)$  for the deterministic complexity,  $R_\delta(f)$  for the bounded two-sided error randomized complexity with error  $0 < \delta < 1/2$ , and  $N(f)$  for the nondeterministic communication complexity. These measures are with respect to the worst-case partition of the inputs of  $f$  between the two parties. For formal definitions, the readers are referred to the book by Kushilevitz and Nisan [6]. Classically, the communication complexity of equality, set disjointness and their variants have been investigated thoroughly. A well-known result is that for set disjointness, denoted by DISJ,  $R_\delta(\text{DISJ}) = \Omega(n)$ , first proved by Kalyanasundaram and Schnitger [1], and simplified by Razborov [2] (see [5] for an information complexity based proof) while  $N(\text{DISJ}) = O(\log n)$ . Strengthening this result to allow a separation on  $R(f)$  from both  $N(f)$  and  $N(\bar{f})$  was resolved only recently by Jayram, Kumar, and Sivakumar [3] using the tribes function (TRIBES) who showed that  $R(\text{TRIBES}) = \Omega(n)$  while  $N(\text{TRIBES}) = N(\bar{\text{TRIBES}}) = O(\sqrt{n} \log n)$ . This is nearly the best possible gap because  $R(f) \leq D(f) \leq N(f) \cdot N(\bar{f})$  as shown by Aho, Ullman, and Yannakakis [15].

The lower bound for TRIBES illustrates a key difficulty in proving communication complexity lower bounds. The standard approach of proving lower bounds for  $R(f)$  is by showing that large rectangles in the function matrix of  $f$  must have large probability of error under some suitable distribution (there are several variants of this approach all of which can be loosely clubbed as the discrepancy bound). This method can at best show a lower bound of  $N(f)$ . Jayram *et al.* [3] overcome this by using a direct-sum argument via information complexity, and then applying the Hellinger

*distance* measure to exploit the rectangular structure inherent in communication protocols. Extending these techniques to handle more general classes of functions has been an open problem.

A natural class of functions generalizing the DISJ and TRIBES examples is the class of *read-once formulae*. Recall that a read-once formula can be identified with a tree in which nodes are labeled by AND/OR gates and leaves are labeled by literals such that every variable appears at most once in the tree. Note that set disjointness is a depth-2 formula while tribes is a depth-3 read-once formula. Read-once formulae have already been extensively studied in the context of decision trees. Analogous to communication complexity,  $DT(f)$  and  $RT_0(f)$  can be defined accordingly [16]. Snir [17] showed that for the function  $f$  defined by a complete binary AND-OR tree  $RT_0(f) = DT(f)^\alpha$ , where  $\alpha = \log_2(1 + \sqrt{33}) - 2 = 0.753\dots$ , and a matching lower bound was shown by Saks and Wigderson [18]. In fact, they conjectured that for every Boolean function  $f$ ,  $RT_0(f) = \Omega((DT(f))^\alpha)$ , a conjecture which is still open. Heiman and Wigderson [19] made some progress on this conjecture by showing that for every read-once formula  $f$ ,  $RT_0(f) = \Omega((DT(f))^{0.51})$ . Building on [18], Heiman, Newman, and Wigderson [20] showed that  $RT_0(f) = \Omega(n/2^d)$  for any function  $f$  that can be computed by depth- $d$  read-once threshold formula (that also allows threshold gates).

In this paper, we consider the randomized communication complexity of general read-once  $AC^0$  formulae. Note that lower bounds on randomized communication complexity imply lower bounds on randomized decision trees, because any randomized decision tree can be easily simulated by a communication protocol in which the currently queried node can be revealed by some player using 1 bit (and the depth of the tree corresponds to the communication cost).

Our main result is that for balanced AND-OR trees  $T$  with  $n$  inputs and depth  $d$ , we show that the communication complexity of the function  $f^T(x, y) = T(x \circ y)$  is  $\Omega(n/4^d)$ . Using this, we show the following lower bound on the communication complexity of general AND-OR trees  $T$  with  $n$  inputs and depth  $d$ .

**Theorem I.1.** *Given an arbitrary AND-OR tree  $T$  of depth  $d$  with  $n$  leaves, we have :*

$$R_\delta(\mathcal{T}) \geq \frac{n}{16^d d!} (1 - 2\sqrt{\delta}) = \Omega(n(1 - 2\sqrt{\delta}) / \exp(d \log d))$$

These results generalize the classical results on the communication complexity of set-disjointness [1], [2] (where  $T$  is an OR -gate) and recent results on the

communication complexity of the TRIBES functions [3] (where  $T$  is a depth-2 read-once formula).

Independent of our work, Leonardos and Saks [21] have informed us that they have obtained a lower bound of  $n/8^d$  for general trees of depth  $d$ .

**Techniques:** In this paper, we extend the information complexity paradigm to handle general read-once formulae. Information theoretic arguments have been implicitly used in previous work [22], [23], [24]. Chakrabarti, Shi, Wirth, and Yao [4] were the first to consider information complexity as a formal resource measure while proving direct sum theorems for two-party simultaneous protocols. Bar-Yossef, Jayram, Kumar, and Sivakumar [5] considered a generalization of this measure for general communication protocols. In particular, they introduced *conditional information complexity* in order to handle non-product distributions that are essential for proving tight lower bounds.

In the information complexity paradigm, the idea is to construct a distribution over inputs and lower bound the entropy of the protocol transcript over this distribution. The information revealed by the transcript about the inputs serves as the natural lower bound on the entropy of the transcript. More precisely, the lower bound on the communication complexity is given by the mutual information  $I(Z : \Pi)$  between the inputs  $Z = (X, Y)$  to the two players and the protocol transcript  $\Pi$  (for now assume that  $\Pi$  is a private coin protocol).

The mutual information  $I(Z : \Pi)$  exhibits a direct-sum property that makes the task of lower bounding it tractable. Let us consider the case of the DISJ function given by OR  $(X \wedge Y)$  where  $X, Y$  are  $n$ -bit inputs,  $\wedge$  is the bit-wise AND. In this case, [5] used a distribution for which each bit  $X_i, Y_i$  of the input  $X_i \wedge Y_i = 0$ , and the different bits are independent of each other. On measuring the mutual information  $I(Z : \Pi | D)$  conditioned on an appropriately chosen  $D$ , the information complexity of the DISJ can be shown to be at least  $n$  times the information complexity of the AND function on two bits. Thus the problem reduces to lower bounding the information complexity of a much simpler function. To show the information complexity lower bound for AND, [5] exploit the rectangle property of communication protocols. Specifically, the Hellinger distance between probability distributions proves very useful in exploiting the rectangle property of communication protocols.

The intuition behind the direct sum for DISJ is that when all the inputs to the OR gate are 0, a protocol has to check every input, since any single input could alter the value of the function. However, this intuition fails already at depth 2, in case of the TRIBES

function. Basically, there are no inputs to the TRIBES function where  $\Omega(n)$  inputs can affect the output. To circumvent this difficulty, [3] introduced the notion of *partial information cost*. Informally, this is a quantity that measures the information revealed by the transcript on only a portion  $(X', Y')$  of the input  $(X, Y)$ . Exactly which portion is measured is determined by the conditioning random variable  $D$ . In fact, the approach is to measure the mutual information with the part of the input  $(X', Y')$  that will turn out to be *irrelevant*: the remaining part of  $(X, Y)$  is sufficient to determine the value of the function. This is counterintuitive, since standard applications of information complexity argue that information about a part of the input needs to be revealed precisely because it might directly determine the value of the function. Nevertheless, the notion is not meaningless: note that the protocol itself does not know about the conditioning random variable and therefore does not know which portion is irrelevant. In [3], an appropriately chosen information complexity of the TRIBES function is reduced to a corresponding cost of a binary AND-OR tree of depth 2. Again, the information cost of the binary AND-OR tree is lower bounded using Hellinger distance.

Along these lines, the natural approach would be to express the information cost of a depth  $d$  AND-OR tree as the direct sum of the cost of binary trees of depth  $d$ . Then show a lower bound on the information cost of the binary tree possibly using Hellinger distances. However, executing this plan poses three main challenges.

Firstly, for the information cost as defined in [3], showing the lower bound on information complexity of a binary tree of depth  $d$  becomes unwieldy. In case of DISJ and TRIBES, the information cost could be expressed in terms Hellinger distance between the distribution of transcripts, when the input was fixed to various values. The fixing of the inputs was crucial in exploiting the properties of Hellinger distance such as the “cut-paste lemma”.

Even for a binary tree of depth 3, as defined in [3], the information cost involves Hellinger distances between distributions over transcripts, where the input itself is not fixed. To circumvent this, we define a new information cost expression where one measures the mutual information of the irrelevant bits conditioned on a particular fixing of the relevant bits. We call this complexity measure the *irrelevant information cost* of the protocol. The notion of *irrelevant information cost* (IRIC) could be of independent interest.

More importantly, unlike the case of DISJ [5] or TRIBES [3] showing the direct sum of information com-

plexity becomes unwieldy. Intuitively, the DISJ function consists of  $n$  completely disjoint AND gates. In case of TRIBES, the function does not decompose in this manner in to disjoint binary trees of depth 2, making the direct sum argument harder. In [3], the information cost expressions are manipulated one level at a time. This approach appears difficult to carry out for general depth, or even for TRIBES function with non-uniform degree. We define a carefully chosen distribution on binary subtrees of an arbitrary non-regular tree. Using this distribution, we are able to perform the direct sum argument for a depth  $d$  tree in one shot.

With this subtle change in the definition of information cost, the natural extension of [3] approach would yield a lower bound of  $1/\exp(\exp(d))$  on the information complexity of a binary tree of depth  $d$ . To obtain a bound that decreases exponential in  $d$ , we alter the input distribution. Roughly speaking, we devise a special gadget distribution  $D$  for the inputs of a binary AND-OR tree such that, the information cost under  $D$  can be easily lower bounded to  $1/\exp(d)$  using Hellinger distances. Since the distribution  $D$  is chosen specifically for this purpose, it does not have the symmetries or the recursive structure available in distribution in [3]. However, by embedding samples from  $D$  at carefully chosen input locations of the bigger AND-OR tree, one produces the required symmetries.

**Organization of this paper:** In Section II, we introduce some preliminaries on AND-OR trees and information complexity. In Section III, we define the “hard” input distributions under which we will measure information complexity. In Section IV, we define the precise version of information complexity that we will use for our lower bounds. In Section V, we derive a lower bound on the information complexity for approximately balanced AND-OR trees in terms of the information complexity of a binary AND-OR tree of the same height. In Section VI, we use the lower bounds for approximately balanced AND-OR trees to prove Theorem I.1 on the communication complexity of arbitrary AND-OR trees. Finally, in Section VII we show a lower bound on the information complexity of binary AND-OR trees.

## II. PRELIMINARIES

In this section, we set up some notation and briefly review the information complexity notions developed in [5], [3].

**AND-OR Trees:** Let  $\mathcal{T}$  denote an (alternating) AND-OR tree with  $n$  leaves and depth at most  $d$ . While all internal nodes of an AND-OR tree  $\mathcal{T}$  is are gates, the  $n$  leaves represent input wires. The read-once formula  $\mathcal{T}(X, Y)$  associated with  $\mathcal{T}$  is obtained by replacing each leaf

$v$  by either  $X_v \vee Y_v$  or  $X_v \wedge Y_v$ , where  $(X_v, Y_v)$  is a distinct pair of Boolean variables, such that the resulting formula has alternating levels of AND and OR gates. Specifically, if the parent of  $v$  is an AND gate, then  $v$  is an OR gate and vice versa. Henceforth, we shall abuse notation and use  $\mathcal{T}$  to denote both the AND-OR tree and the read-once formula  $\mathcal{T}(X, Y)$  computed by it.

Let  $\mathcal{L}(\mathcal{T})$  denote the set of leaves of  $\mathcal{T}$ . For an internal node  $v$ , let  $C(v)$  denote the set of children of  $v$ . For each node  $v$ , let  $\text{ht}(v)$  denote the height of node  $v$ , with leaves being of height 0. Let  $\mathcal{T}_v$  denote the subtree of  $\mathcal{T}$  rooted at  $v$  and the function computed by the subtree.

For each node  $v \notin \mathcal{L}(\mathcal{T})$ , we let  $X_v, Y_v$  denote the vector of inputs to the subtree  $\mathcal{T}_v$ . Formally,

$$X_v = (X_u)_{u \in \mathcal{L}(\mathcal{T}) \cap \mathcal{T}_v} \quad Y_v = (Y_u)_{u \in \mathcal{L}(\mathcal{T}) \cap \mathcal{T}_v}.$$

Let  $Z_v$  denote the pair of inputs  $(X_v, Y_v)$ . Thus the input to the tree is  $Z_{\mathcal{T}} = (X_{\mathcal{T}}, Y_{\mathcal{T}})$ , where  $X_{\mathcal{T}} = (X_v)_{v \in \mathcal{L}(\mathcal{T})}$  and  $Y_{\mathcal{T}} = (Y_v)_{v \in \mathcal{L}(\mathcal{T})}$ . If  $\gamma, \delta$  are partial assignments to disjoint subsets  $S_1, S_2$  of input variables to the tree, we denote by  $\gamma \cup \delta$  the partial assignment extending  $\gamma, \delta$  to the subset  $S_1 \cup S_2$  of input variables to the tree (when there is no risk of confusion, we will write this as  $\gamma\delta$ ).

First, we will show the lower bound on communication complexity for a special case of AND-OR trees defined below,

**Definition II.1** (*c*-balanced AND-OR tree). *For a constant  $0 < c < 1$ , a tree  $\mathcal{T}$  of depth  $d$  is said to be *c*-balanced if*

- Every root to leaf path in  $\mathcal{T}$  is of length  $d$ .
- For every node  $w \in \mathcal{T}$ , and every child  $u \in C(w)$ , we have

$$|\mathcal{L}(\mathcal{T}_u)| \leq (1 - c) \times |\mathcal{L}(\mathcal{T}_w)|$$

The reduction from arbitrary AND-OR trees to *c*-balanced AND-OR trees is described in Claim VI.6. For a *c*-balanced tree, by inverting the function if necessary, we can assume that each leaf is the child of an OR node. Unless otherwise mentioned, we will make this assumption throughout the article, and write  $\mathcal{T}(X, Y) = \mathcal{T}(X \wedge Y)$ .

Due to space constraints, in this version of the paper we demonstrate our argument giving a lower bound for *c*-balanced AND-OR trees on the special case of *regular trees*, defined below.

**Definition II.2** (Regular AND-OR tree). *A tree  $\mathcal{T}$  of depth  $d$  is said to be regular, if the degree of all nodes at a given height are equal.*

**Information Complexity:** Let  $\mu$  be a distribution on the inputs  $(X, Y)$ , denoted by  $(X, Y) \sim \mu$ . We say that  $\mu$  is *product* if  $X$  and  $Y$  are *independent*. Non-product distributions are handled via a new random variable  $D$  such that  $X$  and  $Y$  are independent conditioned on  $D$ . The (*conditional*) *information cost* of a randomized protocol  $\Pi$  under  $(\mu, \nu)$  is defined to be  $I(X, Y : \Pi(X, Y) \mid D, \mathcal{R})$ , where  $(X, Y) \sim \mu, D \sim \nu$  and  $\mathcal{R}$  denotes the public coins. We also allow  $\Pi$  to use private coins, which results in a more robust measure for information complexity. Since  $I(X, Y : \Pi(X, Y) \mid D, \mathcal{R}) \leq H(\Pi(X, Y)) \leq \mathbf{E}|\Pi|$ , the information cost of a correct protocol for a function  $f$  is a lower bound on the communication complexity of  $f$ .

The *Hellinger distance*  $h(P, Q)$  between two probability distributions  $P, Q : \Omega \rightarrow [0, 1]$  is defined by

$$h^2(P, Q) \triangleq \frac{1}{2} \sum_{\omega \in \Omega} (\sqrt{P(\omega)} - \sqrt{Q(\omega)})^2.$$

Hellinger distance satisfies the *triangle inequality*.

Let  $\Pi$  be a randomized private-coin 2-party communication protocol for a function  $f(x, y)$ , with error probability at most  $\delta$ . (For protocols that also use public coins, we will consider the transcript under an arbitrary assignment to the public coins.) For inputs  $(a, b)$  to the players, let  $\Pi(a, b)$  denote the random variable containing the transcript of the communication, and let  $P(a, b)$  be its distribution.

**Proposition II.3** (Soundness of communication protocols). *Let  $f, \Pi, \delta$  and  $P(\cdot, \cdot)$  be as above. Let  $(a, b)$  and  $(a', b')$  be two pairs of inputs. If  $f(a, b) \neq f(a', b')$ , then  $h^2(P(a, b), P(a', b')) \geq 1 - 2\sqrt{\delta}$ .*

The following lemma allows us to transition from information complexity to Hellinger distance:

**Lemma II.4** (Mutual information versus Hellinger distance). *Let  $Z_1$  and  $Z_2$  be random variables, and let  $P_1$  and  $P_2$  be their distributions. Then if  $B$  is a uniformly random bit (independent of the  $Z_i$ ), we have*

$$I(B; Z_B) \geq h^2(P_1, P_2).$$

### III. HARD DISTRIBUTIONS FOR REGULAR TREES

In this section, we will describe a hard distribution of inputs to a *regular* AND-OR tree  $\mathcal{T}$  with OR gates at the bottom layer, and show some useful properties of the distribution. The hard distribution of inputs for general *c*-balanced trees is similar in spirit, and its description is omitted in this extended abstract.

**Definition III.1.** *Let  $v, w \in \mathcal{T}$  be two vertices such that  $w$  is a parent of  $v$  ( $v \in C(w)$ ). For an input  $Z_{\mathcal{T}}$ , the*

vertex  $v$  is said to be relevant under  $Z_{\mathcal{T}}$ , if the output  $\mathcal{T}_v(Z_v)$  of the node  $v$ , fixes the value  $\mathcal{T}_w(Z_w)$  of the parent node  $w$ . Otherwise, the vertex  $v$  is said to be irrelevant under  $Z_{\mathcal{T}}$ .

For an OR gate, a *relevant* child is one that evaluates to 1, thus fixing the value of the OR gate. Similarly, a *relevant* child of an AND gate always evaluates to 0.

Note that on evaluating a *relevant* child of  $w$ , the computation of the output of  $w$  is complete. Thus, intuitively, a hard distribution of inputs must maximize the number of irrelevant children of a node  $v$ . With this intuition, a natural hard distribution can be constructed in a top-down manner as described below.

**Canonical Hard Distribution:** Fix the output of the root to say 0 or 1 arbitrarily. This choice is propagated to the leaves of the tree, while at all times gates the number of irrelevant branches. For example, if an OR gate  $v$  that is set to 0, then all its inputs must necessarily be set to 0 (all children are irrelevant). On the other hand, if the output is set to 1, then one of its children  $s_v$  is *selected* to be *relevant* (equal to 1), while all other children are made irrelevant (set to 0). For AND gates, the propagation is defined along similar lines with the 0 and 1 interchanged.

The node  $s_v$  chosen to be the relevant child of  $v$  will be referred to as the *Selector*. Using the same principle, for a leaf node  $w \in \mathcal{L}(\mathcal{T})$ , if  $w$  is set to 1, then both the inputs  $X_w, Y_w$  are forced to be *irrelevant* (equal to 1). On the other hand, if  $w$  is fixed to 0, then one of the inputs  $X_w, Y_w$  is chosen to be *relevant* (equal to 0), while the other is set to a random bit.

The *Leaf Selector*  $\sigma_w \in \{1, 2\}$  corresponding to the leaf  $w$  encodes the choice of relevant variable among  $X_w$  and  $Y_w$ .

Notice that the above construction is completely determined by the choice of the distribution of selectors and leaf selectors. The hard distribution of inputs constructed in this work is somewhat more involved. We will describe the construction of the hard distribution in four steps.

#### A. Selectors and Leaf Selectors

Formally, for each node  $v \in \mathcal{T}$ , the selector  $s_v \in C(v)$  is a random variable equal to one of its children. The set  $S_v$  consists of all the random variables  $s_u$  for each  $u$  in the subtree rooted at  $v$ .

$$S_v = \{s_u | u \in \mathcal{T}_v\}$$

Similarly, we shall denote  $S_{\mathcal{T}} = \{s_u | u \in \mathcal{T}\}$ . For each leaf node  $w \in \mathcal{L}(\mathcal{T})$ , the *leaf selector*  $\sigma_w$  is chosen uniformly at random from  $\{1, 2\}$ . We shall denote  $\sigma_{\mathcal{T}} =$

$\{\sigma_w | w \in \mathcal{L}(\mathcal{T})\}$ . The distribution of  $S_{\mathcal{T}}$  and  $\sigma_{\mathcal{T}}$  for regular trees is as follows:

**Sampling  $S_{\mathcal{T}}, \sigma_{\mathcal{T}}$  (regular trees):** For each  $u \in \mathcal{T}$  with  $\text{ht}(u) \geq 1$ , set  $s_u$  to be a uniform random element from  $C(u)$ .  
For each leaf  $v \in \mathcal{L}(\mathcal{T})$  of the tree  $\mathcal{T}$ ,  $\sigma_v$  is a random element from  $\{1, 2\}$ .

#### B. Binary Subtrees

For each node  $v$ ,  $\mathbb{T}_v^{bin}$  is a random binary tree rooted at  $v$ . Roughly speaking, we embed the gadget distributions  $D_i$  described in the next section, in to the leaves of the random binary subtree  $\mathbb{T}_v^{bin}$ . Here, we will distribution of  $\mathbb{T}_v^{bin}$  conditioned on the choice of the selectors  $S_v$ . The random tree  $\mathbb{T}_v^{bin}$  will be constructed so as to satisfy the following property,

**Property III.2.** For each vertex  $u \in \mathbb{T}_v^{bin}$ , the selector  $s_u \in \mathbb{T}_v^{bin}$ .

A random binary tree  $\mathbb{T}_v^{bin}$  satisfying the above property is generated as follows,

**Sampling  $\mathbb{T}_v^{bin}$  given  $S_v$  (regular trees):**

- Sample the “pseudo-selector”  $s'_v$  uniformly at random from  $C(v) - \{s_v\}$ .
- Recursively generate binary trees  $\mathbb{T}_{s_v}^{bin}, \mathbb{T}_{s'_v}^{bin}$  given  $S_{s_v}$  and  $S_{s'_v}$  respectively.
- The tree  $\mathbb{T}_v^{bin}$  consists of  $\{v, \mathbb{T}_{s_v}^{bin}, \mathbb{T}_{s'_v}^{bin}\}$ .

#### C. Gadget input distribution $D_i$ to binary trees

The hard distribution of inputs consists embeddings of a “gadget” distribution in to certain carefully chosen random binary subtrees of  $\mathcal{T}$ .

Let  $\mathbb{T}$  be the complete binary tree of height  $i$  with OR gates at the bottom layer. We will think of  $\mathbb{T}$  as defining a function from  $\{0, 1\}^{2^i} \times \{0, 1\}^{2^i}$  to  $\{0, 1\}$  defined by  $\mathbb{T}(X, Y)$  for  $\mathbb{T}(X \wedge Y)$ .  $D_i$  is a distribution over inputs  $X, Y$  to  $\mathbb{T}$  that satisfies the following property:

**Property III.3.** If  $v_0$  is an AND gate, then the output of  $\mathbb{T}$  under  $D_i$  is always 1, else it is always 0.

The details of construction of the gadget distributions is described in Section VII.

#### D. Input Distribution

Let  $X_{\mathcal{T}} = (X_v)_{v \in \mathcal{L}(\mathcal{T})}$  and  $Y_{\mathcal{T}} = (Y_v)_{v \in \mathcal{L}(\mathcal{T})}$  be the input variables to the AND-OR tree, and let  $Z_{\mathcal{T}} = (X_{\mathcal{T}}, Y_{\mathcal{T}})$ . For a node  $v$ , let  $Z_v^{bin} = (X_v^{bin}, Y_v^{bin})$  denote the inputs at the leaves of  $\mathbb{T}_v^{bin}$ . Hence  $Z_v - Z_v^{bin}$  will denote the remaining inputs in the subtree  $\mathcal{T}_v$  rooted at  $v$ .

**Input Distribution given  $S_{\mathcal{T}}, \sigma_{\mathcal{T}}$** 

Define distributions  $\mathcal{D}_v^{\text{Rel}}$  and  $\mathcal{D}_v^{\text{Irrel}}$  for the inputs  $Z_v = (X_v, Y_v)$  recursively as follows:

- If  $v$  is a leaf then  $\mathcal{D}_v^{\text{Rel}}$  is supported on the input  $(1, 1)$ , while  $\mathcal{D}_v^{\text{Irrel}}$  is defined as follows:

$$(X_v, Y_v) = \begin{cases} (0, \text{random bit}) & \text{if } \sigma_v = 1 \\ (\text{random bit}, 0) & \text{if } \sigma_v = 2 \end{cases}$$

- If  $v$  is an internal gate then,
  - Irrelevant Branch  $\mathcal{D}_v^{\text{Irrel}}$ : Generate  $Z_{s_v}$  from the distribution  $\mathcal{D}_{s_v}^{\text{Rel}}$ , and  $Z_u$  according to  $\mathcal{D}_u^{\text{Irrel}}$  for all other  $u \in C(v) - \{s_v\}$ . Then the input  $Z_v$  is the union of these inputs, i.e.

$$Z_v = \bigcup_{u \in C(v), u \neq s_v} Z_u \cup Z_{s_v}$$

- Relevant Branch  $\mathcal{D}_v^{\text{Rel}}$ :
  - \* Sample  $\mathbb{T}_v^{\text{bin}}$  by generating the pseudo-selectors needed.
  - \* Generate  $Z_v^{\text{bin}}$  according to the distribution  $\mathcal{D}_{\text{ht}(v)}$ .
  - \* For every node  $w \in T_v - \mathbb{T}_v^{\text{bin}}$  such that its parent belongs to  $\mathbb{T}_v^{\text{bin}}$ , generate  $Z_w$  from  $\mathcal{D}_w^{\text{Irrel}}$ .

The “hard” distribution  $\mathcal{D}_{\mathcal{T}}$  on  $\{0, 1\}^n \times \{0, 1\}^n$  is given by  $\mathcal{D}_u^{\text{Irrel}}$  where  $u$  is the root of the tree  $\mathcal{T}$ .

**Observation III.4.** The following properties hold for the distributions  $\mathcal{D}_v^{\text{Rel}}$  defined above :

- For an AND gate  $v$ , the subtree  $\mathcal{T}_v$  evaluates to 1 on  $\mathcal{D}_v^{\text{Rel}}$  and 0 on  $\mathcal{D}_v^{\text{Irrel}}$ .
- For an OR gate  $v$ , the subtree  $\mathcal{T}_v$  evaluates to 0 on  $\mathcal{D}_v^{\text{Rel}}$  and 1 on  $\mathcal{D}_v^{\text{Irrel}}$ .
- For a node  $v$ , if  $Z_v$  is generated from  $\mathcal{D}_v^{\text{Irrel}}$ , then the output of AND-OR tree  $\mathcal{T} - \mathcal{T}_v$  on input  $Z_{\mathcal{T} - \mathcal{T}_v}$  is equal to output of  $\mathcal{T}$  on  $Z_{\mathcal{T}}$ . Roughly speaking, dropping an irrelevant branch does not change the output value of the tree.

Let  $S'_{\mathcal{T}}$  denote the set of pseudo-selector random variables sampled in the above procedure. Clearly,  $S'_{\mathcal{T}}$  itself is a random subset of vertices of  $\mathcal{T}$  depending on  $S_{\mathcal{T}}$ . Let us denote by  $S'_v$  the restriction of the set  $S'_{\mathcal{T}}$  to the subtree  $\mathcal{T}_v$ .

**E. Properties of Selectors  $S_{\mathcal{T}}$  and Binary Trees  $\mathbb{T}^{\text{bin}}$** 

We now state some crucial properties of the selector and binary tree random variables that we just defined. The generalizations of these distributions for the case of  $c$ -balanced AND-OR trees are designed so as to ensure that these crucial properties still hold.

**Claim III.5.** For every node  $v \in \mathcal{T}$  and two of its children  $u_1, u_2 \in C(v)$

$$\begin{aligned} \Pr \left[ s_v = u_1, s'_v = u_2 \mid S_{\mathcal{T}} - \{s_v\}, S'_{\mathcal{T}} - \{s'_v\} \right] \\ = \Pr[s_v = u_2, s'_v = u_1 \mid S_{\mathcal{T}} - \{s_v\}, S'_{\mathcal{T}} - \{s'_v\}] \end{aligned}$$

**Lemma III.6.** Consider a node  $w \in \mathbb{T}_v^{\text{bin}}$  and two of its children  $u_1, u_2$  in  $\mathbb{T}_v^{\text{bin}}$ . Conditioned on  $\mathbb{T}_v^{\text{bin}}$ , the selector  $s_w$  is uniformly distributed among  $\{u_1, u_2\}$  irrespective of the choice of every other selector. Formally,

$$\Pr[s_w = u_1 \mid \mathbb{T}_v^{\text{bin}}, S_{\mathcal{T}} - \{s_w\}] = \Pr[s_w = u_2 \mid \mathbb{T}_v^{\text{bin}}, S_{\mathcal{T}} - \{s_w\}] = \frac{1}{2}$$

**F. Maximally Irrelevant Leaves**

As described earlier, the choice of selectors  $S_{\mathcal{T}}$  determines the vertices that will be relevant. Therefore, we make the following definition:

**Definition III.7** (Maximally Irrelevant Leaves). Let  $v \in \mathcal{T}_u \cap \mathcal{L}(T)$ . Let  $\mathcal{P}_v = \{u_0 = u, u_1, \dots, u_t = v\}$  denote the unique path from  $u$  to node  $v \in \mathcal{T}_u$ . An assignment to  $S_u$  is said to make  $v$  maximally irrelevant if for each  $0 \leq i < t$ ,  $s_{u_i} \neq u_{i+1}$ . For the sake of brevity, we shall denote this by  $v \in \text{Irrel}(S_u)$ . Denote by  $\text{Rel}(S_{\mathcal{T}})$  the set of leaves of  $\mathcal{T}$  that are not in the set  $\text{Irrel}(S_{\mathcal{T}})$ .

**Lemma III.8.** Given  $S_v$ , the random binary subtree  $\mathbb{T}_v^{\text{bin}}$  can equivalently be generated as follows:

- Pick a leaf  $w \in \text{Irrel}(S_v)$  uniformly at random. Let  $\mathcal{P} = \{u_0 = v, u_1, \dots, u_t = w\}$  denote the unique path from  $v$  to node  $w$ .
- Generate binary subtrees  $\mathbb{T}_{s_{u_0}}^{\text{bin}}, \mathbb{T}_{s_{u_1}}^{\text{bin}}, \dots, \mathbb{T}_{s_{u_{t-1}}}^{\text{bin}}$ , and take their union with  $\mathcal{P}$  to obtain  $\mathbb{T}_v^{\text{bin}}$ .

*Proof:* Consider a sample  $\mathbb{T}$  of the random binary tree  $\mathbb{T}_v^{\text{bin}}$ . The tree  $\mathbb{T}$  has a unique vertex  $w \in \text{Irrel}(S_{\mathcal{T}})$ . Consider the path  $\mathcal{P} = \{u_0 = v, u_1, \dots, u_t = w\}$  from  $v$  to  $w$ . Let  $\mathbb{T}_0, \dots, \mathbb{T}_{d-1}$  denote the subtrees of  $\mathbb{T}$  rooted at  $s_{u_i}$ . By definition of the distribution for  $\mathbb{T}_v^{\text{bin}}$ , it is clear that,

$$\begin{aligned} \Pr[\mathbb{T}^{\text{bin}} = \mathbb{T}] &= \prod_{i=0}^{d-1} \left( \Pr[s'_{u_i} = u_{i+1}] \Pr[\mathbb{T}_i] \right) \\ &= \prod_{i=0}^{d-1} \left( \frac{1}{|C(u_i)| - 1} \right) \left( \prod_{i=0}^{d-1} \Pr[\mathbb{T}_i] \right), \end{aligned}$$

where we used that  $\Pr[s'_{u_i} = u_{i+1}] = \frac{1}{|C(u_i)| - 1}$  since  $s'_{u_i}$  is a uniform random element in  $C(u_i) - \{s_{u_i}\}$ . By regularity, observe that  $\prod_{i=0}^{d-1} (|C(u_i)| - 1)$  is the number of maximally irrelevant leaves. Hence,  $\mathbb{T}_v^{\text{bin}}$  can be equivalently generated by picking a leaf  $w \in \text{Irrel}(S_v)$

uniformly at random and generating binary subtrees  $\mathbb{T}_{s_{u_0}}^{bin}, \mathbb{T}_{s_{u_1}}^{bin}, \dots, \mathbb{T}_{s_{u_{t-1}}}^{bin}$  ■

Let  $Z_{\text{Irrel}(S_{\mathcal{T}})}$  denote the restriction of  $Z_{\mathcal{T}}$  to coordinates corresponding to leaves that are made maximally irrelevant by the selector choice  $S_{\mathcal{T}}$ . Formally,

$$Z_{\text{Irrel}(S_{\mathcal{T}})} = (X_v, Y_v)_{v \in \text{Irrel}(S_{\mathcal{T}})}$$

For the sake of brevity, we shall write  $Z^{\text{Irrel}}$  instead of  $Z_{\text{Irrel}(S_{\mathcal{T}})}$  when the choice of  $S_{\mathcal{T}}$  is clear from the context. Further, let  $Z^{\text{Rel}} = Z_{\mathcal{T}} - Z^{\text{Irrel}}$ .

**Lemma III.9.** *Conditioned on  $S_{\mathcal{T}}, S'_{\mathcal{T}}$ , the random variables  $\{Z_v | v \in \text{Irrel}(S_{\mathcal{T}})\}$  are independent of each other.*

*Proof:* Consider a maximally irrelevant leaf  $v \in \text{Irrel}(S_{\mathcal{T}})$ . For every node  $w$  along the path from the root  $\rho$  to  $v$ , the recursive procedure always takes the irrelevant branch, thus generating  $\mathcal{D}_w^{\text{Irrel}}$ . Independent of all other inputs,  $Z_v$  is generated from the distribution  $\mathcal{D}_v^{\text{Irrel}}$  using  $\sigma_v$  at the base case of the recursion. Hence the random variables  $\{Z_v | v \in \text{Irrel}(S_{\mathcal{T}})\}$  are all independent of each other. ■

#### Special nodes:

Let us call a node  $v$  to be Special, if we sample the binary tree  $\mathbb{T}_v^{bin}$  in the recursive procedure to generate inputs. The set of Special nodes is a random set completely determined by the choice of  $S_{\mathcal{T}}$  and  $S'_{\mathcal{T}}$ . Hence we shall write  $\text{Special}(S_{\mathcal{T}}, S'_{\mathcal{T}})$  to denote the set of Special nodes.

**Observation III.10.** *Let  $\mathcal{P}_v = \{u_0 = \rho, \dots, u_t = v\}$  denote a path from the root  $\rho$  to a vertex  $u_t$  in the AND-OR tree  $\mathcal{T}$ . If  $s_{u_i} \neq u_{i+1}$  for all  $i < t-1$ , and  $s_{u_{t-1}} = u_t$  then  $u_t$  is a Special node.*

Define  $Z^{\text{Special}}$  as

$$Z^{\text{Special}} = (X_v, Y_v)_{v \in \mathbb{T}_u^{bin}, u \in \text{Special}(S_{\mathcal{T}}, S'_{\mathcal{T}})}$$

Let  $Z_u^{\text{Special}}$  denote the Special inputs within the subtree  $\mathcal{T}_u$ . Formally,

$$Z_u^{\text{Special}} = (X_v, Y_v)_{v \in \mathbb{T}_w^{bin}, w \in \mathcal{T}_u \cap \text{Special}(S_{\mathcal{T}}, S'_{\mathcal{T}})}$$

Every input in  $Z^{\text{Special}}$  is generated from one of the tailor-made distributions  $\{D_i\}$ . Every input  $Z_v$  not in  $Z^{\text{Special}}$  is generated through the base case of the above recursive definition (using the leaf selector  $\sigma_v$ ).

Accounting for all the shared randomness used in generating a sample from the distribution  $\mathcal{D}_v^{\text{Irrel}}$ , we make the following observation.

**Observation III.11.** *For any node  $v \in \mathcal{T}$ , using shared randomness  $\mathcal{R}_v = \{S_v, S'_v, \sigma_v, Z_v^{\text{Special}}\}$  the two players*

*can sample their inputs  $X_v, Y_v$  from the  $\mathcal{D}_v^{\text{Irrel}}$  and  $\mathcal{D}_v^{\text{Rel}}$  without any communication.*

## IV. IRRELEVANT INFORMATION COST

Let  $\Pi$  be a protocol computing  $\mathcal{T}(X \wedge Y)$  for a  $c$ -balanced AND-OR tree  $\mathcal{T}$  of depth  $d$ . To lower bound the communication between the players in the protocol  $\Pi$ , we will make use of the following information cost.

**Definition IV.1.** *The irrelevant information cost  $\text{IRIC}(\Pi, \mathcal{T})$  for the protocol  $\Pi$  is defined as*

$$\text{IRIC}(\Pi, \mathcal{T}) = I(Z_{\mathcal{T}}^{\text{Irrel}} : \Pi | Z_{\mathcal{T}}^{\text{Special}}, S_{\mathcal{T}}, S'_{\mathcal{T}}, \sigma_{\mathcal{T}}, \mathcal{R}) \quad (1)$$

where  $\Pi$  denotes transcript of the protocol executed with shared randomness  $\mathcal{R}$  over the input  $Z_{\mathcal{T}}$  generated using the choices  $S_{\mathcal{T}}, S'_{\mathcal{T}}$  and  $\sigma_{\mathcal{T}}$ . Further, the irrelevant information complexity  $\text{IRIC}(\mathcal{T})$  is defined as the minimum of  $\text{IRIC}(\Pi, \mathcal{T})$  over all randomized  $\delta$ -error protocols for  $\mathcal{T}$ .

$$\text{IRIC}(\mathcal{T}) = \min_{\Pi} \text{IRIC}(\Pi, \mathcal{T}) \quad (2)$$

Clearly,  $\text{IRIC}(\Pi, \mathcal{T}) \leq H(\Pi)$  where  $\Pi$  denotes the random variable equal to the transcript of the protocol. Consequently,  $\text{IRIC}(\Pi, \mathcal{T})$  serves as a lower bound on the communication complexity of the protocol  $\Pi$ .

**Observation IV.2.** *By independence of  $Z_v$  for  $v \in \text{Irrel}(S_{\mathcal{T}})$  (Lemma III.9),*

$$\text{IRIC}(\Pi, \mathcal{T}) \geq \sum_{v \in \text{Irrel}(S_{\mathcal{T}})} I(Z_v : \Pi | Z^{\text{Special}}, S_{\mathcal{T}}, S'_{\mathcal{T}}, \sigma_{\mathcal{T}}, \mathcal{R})$$

### A. Information Cost of a Binary Tree

The input distribution constructed in Section III for an arbitrary AND-OR tree reduces to the distribution given in Table I for the case of a balanced binary tree  $\mathcal{T}$ .

The irrelevant information cost of a protocol  $\Pi^{\text{bin}}$  computing the function  $\mathbb{T}$  while using shared randomness  $\mathcal{R}^{\text{bin}}$  reduces to the following :

$$\text{IRIC}(\Pi^{\text{bin}}, \mathbb{T}) = I(Z_{\nu} : \Pi^{\text{bin}} | Z_{\mathcal{T}-\{\nu\}}, S_{\mathcal{T}}, \sigma_{\mathcal{T}}, \mathcal{R}^{\text{bin}}) \quad (3)$$

Here we used the observation that  $S'_{\mathcal{T}}$  is completely fixed by the assignment to  $S_{\mathcal{T}}$ .

## V. SIMULATING A BINARY TREE

In this section, we will show the following relation between the information cost of a general tree and that of a complete binary tree. In the next section we will apply this theorem to a  $c$ -balanced tree  $\mathcal{T}$ .

**Input Distribution for a binary tree**

- For each vertex  $v \in T$ , set the selector  $s_v$  to be one of its children  $u \in C(v)$  chosen uniformly at random. Let  $S_T$  denote the set of all selectors in  $T$ .
- For each leaf  $v \in T$ , sample the leaf selector  $\sigma_v$  distributed uniformly on  $\{1, 2\}$ .
- Let  $\nu$  denote the unique node in  $\text{Irrel}(S_T)$ . Generate  $Z_\nu = (X_\nu, Y_\nu)$  as follows:

$$(X_\nu, Y_\nu) = \begin{cases} (0, \text{random bit}) & \text{if } \sigma_\nu = 1 \\ (\text{random bit}, 0) & \text{if } \sigma_\nu = 2 \end{cases}$$

- Let  $\mathcal{P} = \{u_0 = \rho, u_1, \dots, u_d = \nu\}$  denote the path from the root  $\rho$  of  $T$  to the leaf  $\nu$ . For each  $i$ , generate  $Z_{s_{u_i}}$  from the distribution  $D_i$ .

The input is  $Z_T$  is given by

$$Z_T = \cup_{i=0}^{d-1} Z_{s_{u_i}} \cup Z_\nu$$

Table I  
INPUT DISTRIBUTION FOR A BINARY TREE

**Theorem V.1.** For every depth  $d$  AND-OR tree  $T$ ,

$$\text{IRIC}(T) \geq \left( \min_{S_T} |\text{Irrel}(S_T)| \right) \cdot \text{IRIC}(T)$$

where  $T$  is the binary AND-OR tree of depth  $d$ .

*Proof:* Given a protocol  $\Pi$  for  $T$ , we will construct a protocol  $\Pi^{\text{bin}}$  for the complete binary tree  $T$  of depth  $d$ .

**Protocol  $\Pi^{\text{bin}}$**

*Input:* The two players get  $\tilde{X}, \tilde{Y} \in \{0, 1\}^{2^d}$  respectively. Let  $\tilde{Z} = (\tilde{X}, \tilde{Y})$ .

Using shared randomness, they sample the following:

- Selectors for the tree  $T$  according to the distribution  $S_T$ . Let  $\hat{S}_T$  denote the sampled selector values.
- A binary subtree  $T^{\text{bin}}$  of the root  $\rho$ , from the distribution  $(T^{\text{bin}} | \hat{S}_T)$ .
- Leaf selectors  $\sigma_{T-T^{\text{bin}}}$  for leaves outside the binary tree  $T^{\text{bin}}$ .
- For each  $u \notin T^{\text{bin}}$  such that  $u \in C(w)$  for  $w \in T^{\text{bin}}$ ,

- Generate  $Z_u$  from the distribution  $D_u^{\text{lrrel}}$  using shared randomness. Specifically, other than  $S_u$  and  $\sigma_u$ , the shared randomness required would be  $\mathcal{R}_u = \{S'_u, Z_u^{\text{Special}}\}$ .

Denote by  $S'_{T-T^{\text{bin}}}, Z_{T-T^{\text{bin}}}^{\text{Special}}, S_{T-T^{\text{bin}}}, Z_{T-T^{\text{bin}}}$  the union over all such  $u$  of  $S'_u, Z_u^{\text{Special}}, S_u, Z_u$ .

Execute the protocol  $\Pi$  on the union of input  $\tilde{Z}$  and  $Z_{T-T^{\text{bin}}}$ . Let us denote by  $\mathcal{R}_T$ , the shared randomness that may be used by the protocol  $\Pi$ .

**Correctness:** Consider a node  $u \notin T^{\text{bin}}$  but directly attached to  $T^{\text{bin}}$ . Formally, let  $u \notin T^{\text{bin}}$  such that  $u \in C(w)$ , for  $w \in T^{\text{bin}}$ . The inputs  $Z_u$  are generated from the distribution  $D_u^{\text{lrrel}}$ . By Item 3 of Observation III.4, every such node  $u$  does not affect the output of the function  $\mathcal{T}$ . In other words, dropping the subtree  $T_u$  does not alter the value of the function. Hence, the above protocol is a randomized  $\delta$ -error protocol for computing  $T(\tilde{X} \wedge \tilde{Y})$ .

For the sake of clarity, we are using the tilde superscript to differentiate random variables associated with the binary tree  $T$  from the others. For instance, a selector  $\tilde{s}_v \in \tilde{S}_T$  picks just one of two children of  $v \in T$ .

**Information Cost:** By Equation (3), the information cost of protocol  $\Pi^{\text{bin}}$  is given by:

$$\text{IRIC}(\Pi^{\text{bin}}, T) = I(\tilde{Z}_\nu : \Pi^{\text{bin}} | \tilde{Z}_{T-\{\nu\}}, \tilde{S}_T, \tilde{\sigma}_T, \mathcal{R}^{\text{bin}}) \quad (4)$$

The shared randomness used by the protocol  $\Pi^{\text{bin}}$  is given by:

$$\mathcal{R}^{\text{bin}} = \{S_T, T^{\text{bin}}, \sigma_{T-T^{\text{bin}}}, S'_{T-T^{\text{bin}}}, Z_{T-T^{\text{bin}}}^{\text{Special}}, \mathcal{R}_T\}$$

Observe that  $\tilde{\sigma}_T$  has the same distribution as  $\sigma_{T^{\text{bin}}}$ . Formally, we write  $\tilde{\sigma}_T \cup \sigma_{T-T^{\text{bin}}} \sim \sigma_T$  where  $\sim$  denotes that the random variables are identically distributed. Rewriting the information cost expression, we get  $\text{IRIC}(\Pi^{\text{bin}}, T) = I(\tilde{Z}_\nu : \Pi | G)$ , where  $G$  denotes the collection

$$\{\tilde{Z}_{T-\{\nu\}}, \tilde{S}_T, \sigma_T, S_T, T^{\text{bin}}, S'_{T-T^{\text{bin}}}, Z_{T-T^{\text{bin}}}^{\text{Special}}, \mathcal{R}_T\}$$

Let us fix a choice of the binary tree  $T^{\text{bin}}$ . By Lemma III.6, we know the following facts:

- For a node  $v \in T^{\text{bin}}$  with two children  $u_1, u_2$  inside  $T^{\text{bin}}$ , the random variable  $(s_v | T^{\text{bin}})$  is uniform over the set  $\{u_1, u_2\}$ . In other words, conditioned on the binary tree  $T^{\text{bin}}$ , the selector random variables  $S_{T^{\text{bin}}}$  are distributed like the selectors in a binary AND-OR tree.
- Conditioned on  $T^{\text{bin}}$ , the random variables  $S_{T-T^{\text{bin}}}$  and  $S_{T^{\text{bin}}}$  are independent of each other.

Thus one can write

$$\begin{aligned} (S_T, T^{\text{bin}}) &\sim ((S_T | T^{\text{bin}}), T^{\text{bin}}) \\ &\sim \left( ((S_{T^{\text{bin}}} | T^{\text{bin}}) \cup (S_{T-T^{\text{bin}}} | T^{\text{bin}})), T^{\text{bin}} \right) \end{aligned}$$

Let us rewrite the shared randomness  $\mathcal{R}^{\text{bin}}$  using the above equivalence,  $\mathcal{R}^{\text{bin}} =$

$$\{T^{\text{bin}}, (S_{T^{\text{bin}}} | T^{\text{bin}}), (S_{T-T^{\text{bin}}} | T^{\text{bin}}), \sigma_{T-T^{\text{bin}}}, S'_{T-T^{\text{bin}}}, Z_{T-T^{\text{bin}}}^{\text{Special}}, \mathcal{R}_T\}$$

Conditioned on  $T^{\text{bin}}$ , we claim that the selectors inside  $S_{T^{\text{bin}}}$  do not affect the execution of the simulation.

Once the tree  $\mathbb{T}^{bin}$  is fixed, the players use the given input  $\tilde{Z}$  for the inputs at leaves of  $\mathbb{T}^{bin}$ . The inputs  $Z_v$  for leaves  $v \in \mathcal{L}(\mathcal{T} - \mathbb{T}^{bin})$  outside the binary tree are all independent of  $S_{\mathbb{T}^{bin}}$ . Thus conditioned on the binary tree  $\mathbb{T}^{bin}$ , the protocol transcript  $\Pi^{bin}$  is independent of  $S_{\mathbb{T}^{bin}}$ . Hence, we can drop  $(S_{\mathbb{T}^{bin}} | \mathbb{T}^{bin})$  from the conditioning altogether to obtain,  $\text{IRIC}(\Pi^{bin}, \mathbb{T}) =$

$$I(\tilde{Z}_\nu : \Pi | \tilde{Z}_{\mathbb{T}-\{\nu\}}, \tilde{S}_{\mathbb{T}}, \sigma_{\mathbb{T}}, \mathbb{T}^{bin}, (S_{\mathbb{T}-\mathbb{T}^{bin}} | \mathbb{T}^{bin}), S'_{\mathbb{T}-\mathbb{T}^{bin}}, Z_{\mathbb{T}-\mathbb{T}^{bin}}^{\text{Special}}, \mathcal{R}_{\mathbb{T}})$$

Let us fix a binary tree  $\mathbb{T}^{bin}$ . Fix a vertex  $v \in \mathbb{T}^{bin}$  with two children  $u_1, u_2$  in  $\mathbb{T}^{bin}$ . The selector  $\tilde{s}_v \in \tilde{S}_{\mathbb{T}}$  uniformly at random one of the two children. By Lemma III.6, the distribution of the selector  $s_v$  conditioned on  $s_v \in \{u_1, u_2\}$  is also uniform over  $u_1, u_2$ , irrespective of the choice of every other selector. Formally, we have

$$\tilde{S}_{\mathbb{T}} \sim ((S_{\mathbb{T}^{bin}} | \mathbb{T}^{bin})) \quad (5)$$

Further we know that conditioned on  $\mathbb{T}^{bin}$ ,  $S_{\mathbb{T}^{bin}}$  is independent of  $S_{\mathbb{T}-\mathbb{T}^{bin}}$ . This tells us that the following random variables are identically distributed :

$$\begin{aligned} & (\tilde{S}_{\mathbb{T}} \cup (S_{\mathbb{T}-\mathbb{T}^{bin}} | \mathbb{T}^{bin})) \\ & \sim ((S_{\mathbb{T}^{bin}} | \mathbb{T}^{bin}) \cup (S_{\mathbb{T}-\mathbb{T}^{bin}} | \mathbb{T}^{bin})) \sim (S_{\mathbb{T}} | \mathbb{T}^{bin}) \end{aligned}$$

Substituting in the information expression we get,  $\text{IRIC}(\Pi^{bin}, \mathbb{T}) =$

$$I(\tilde{Z}_\nu : \Pi | \tilde{Z}_{\mathbb{T}-\{\nu\}}, \sigma_{\mathbb{T}}, \mathbb{T}^{bin}, (S_{\mathbb{T}} | \mathbb{T}^{bin}), S'_{\mathbb{T}-\mathbb{T}^{bin}}, Z_{\mathbb{T}-\mathbb{T}^{bin}}^{\text{Special}}, \mathcal{R}_{\mathbb{T}})$$

Now we will interchange the conditioning  $(\mathbb{T}^{bin}, (S_{\mathbb{T}} | \mathbb{T}^{bin})) \sim ((\mathbb{T}^{bin} | S_{\mathbb{T}}), S_{\mathbb{T}})$  in the above information expression, and get  $\text{IRIC}(\Pi^{bin}, \mathbb{T}) =$

$$I(\tilde{Z}_\nu : \Pi | \tilde{Z}_{\mathbb{T}-\{\nu\}}, \sigma_{\mathbb{T}}, S_{\mathbb{T}}, (\mathbb{T}^{bin} | S_{\mathbb{T}}), S'_{\mathbb{T}-\mathbb{T}^{bin}}, Z_{\mathbb{T}-\mathbb{T}^{bin}}^{\text{Special}}, \mathcal{R}_{\mathbb{T}})$$

Conditioned on  $S_{\mathbb{T}}$ , the tree  $\mathbb{T}^{bin}$  is generated by recursively sampling the pseudo-selectors for every vertex  $v \in \mathbb{T}^{bin}$ . From Lemma III.8, an equivalent sampling procedure would be the following: Choose a path  $\mathcal{P} = \{u_0 = \rho, \dots, u_d = \nu\}$  from the root  $\rho$  of  $\mathcal{T}$  to uniformly random node  $\nu$  in  $\text{lrrel}(S_{\mathbb{T}})$ , and pick binary trees  $\mathbb{T}_1^{bin}, \mathbb{T}_2^{bin}, \dots, \mathbb{T}_{d-1}^{bin}$  rooted at  $u_1, u_2, \dots, u_{d-1}$  respectively. The choice of  $\mathbb{T}_i^{bin}$  is given by the choice of pseudoselectors starting from vertex  $s_{u_i}$  and proceeding downwards. Let  $S'_i$  denote the set of pseudoselectors generated for the subtree  $\mathbb{T}_i^{bin}$ . Conditioning on  $\mathbb{T}^{bin}$  is equivalent to the choice of  $\mathcal{P}$  and  $S'_1, S'_2, \dots, S'_{d-1}$ . We

rewrite the above information expression again and get that  $\text{IRIC}(\Pi^{bin}, \mathbb{T}) =$

$$I(\tilde{Z}_\nu : \Pi | \tilde{Z}_{\mathbb{T}-\{\nu\}}, \sigma_{\mathbb{T}}, S_{\mathbb{T}}, \{\mathcal{P}, S'_1, S'_2, \dots, S'_{d-1}\}, S'_{\mathbb{T}-\mathbb{T}^{bin}}, Z_{\mathbb{T}-\mathbb{T}^{bin}}^{\text{Special}}, \mathcal{R}_{\mathbb{T}})$$

Observe that,

$$\tilde{Z}_{\mathbb{T}-\nu} = \cup_{i=1}^{d-1} \tilde{Z}_{s_{u_i}}$$

where each  $\tilde{Z}_{s_{u_i}}$  is generated from  $D_{\text{ht}(s_{u_i})}$ .

Now, let us consider sampling the input  $Z_{\mathcal{T}}$  for the AND-OR tree  $\mathcal{T}$ , given the choice of selectors  $S_{\mathcal{T}}$ . Given the selectors  $S_{\mathcal{T}}$ , by Observation III.10, each of the nodes  $s_{u_1}, s_{u_2}, \dots, s_{u_{d-1}}$  are Special nodes. For each of these Special nodes  $s_{u_i}$ , one would generate a binary tree  $\mathbb{T}_i^{bin}$  and an input  $Z_{s_{u_i}}^{\text{Special}}$  from the distribution  $D_{h(s_{u_i})}$ . In order to generate the binary trees  $\mathbb{T}_i^{bin}$ , the players would generate pseudo-selectors  $S'_{\mathbb{T}_i^{bin}}$  which is identically distributed to  $S'_i$  defined above, i.e.,

$$S'_{\mathbb{T}-\mathbb{T}^{bin}} \cup (\cup_{i=1}^{d-1} S'_i) \sim S'_{\mathcal{T}}. \quad (6)$$

Further,

$$\tilde{Z}_{\mathbb{T}-\nu} \sim \cup_{i=1}^{d-1} \tilde{Z}_{s_{u_i}} \sim \cup_{i=1}^{d-1} Z_{s_{u_i}}^{\text{Special}} \sim Z_{\mathbb{T}^{bin}}^{\text{Special}},$$

Therefore we have

$$(\tilde{Z}_{\mathbb{T}-\nu} \cup Z_{\mathbb{T}-\mathbb{T}^{bin}}^{\text{Special}}) \sim Z_{\mathcal{T}}^{\text{Special}} \quad (7)$$

Substituting the identities 6,7 back in the information expression one gets,

$$\text{IRIC}(\Pi^{bin}, \mathbb{T}) = I(\tilde{Z}_\nu : \Pi | \mathcal{P}, \sigma_{\mathcal{T}}, S_{\mathcal{T}}, S'_{\mathcal{T}}, Z_{\mathcal{T}}^{\text{Special}}, \mathcal{R}_{\mathcal{T}})$$

Expanding the above information expression along the choice of the vertex  $\nu \in \text{lrrel}(S_{\mathcal{T}})$  (the vertex  $\nu$  fixes the path  $\mathcal{P}$ ).

$$\begin{aligned} \text{IRIC}(\Pi^{bin}, \mathbb{T}^{bin}) &= I(Z_\nu : \Pi | \mathcal{P}, Z_{\mathcal{T}}^{\text{Special}}, S_{\mathcal{T}}, S'_{\mathcal{T}}, \sigma_{\mathcal{T}}, \mathcal{R}_{\mathcal{T}}) \\ &= \mathbf{E}_{S_{\mathcal{T}}^*} \left[ I(Z_\nu : \Pi | \mathcal{P}, Z_{\mathcal{T}}^{\text{Special}}, S_{\mathcal{T}} = S_{\mathcal{T}}^*, S'_{\mathcal{T}}, \sigma_{\mathcal{T}}, \mathcal{R}_{\mathcal{T}}) \right] \\ &= \mathbf{E}_{S_{\mathcal{T}}^*} \left[ \frac{1}{|\text{lrrel}(S_{\mathcal{T}}^*)|} \sum_{\nu \in \text{lrrel}(S_{\mathcal{T}})} I(Z_\nu : \Pi | Z_{\mathcal{T}}^{\text{Special}}, S_{\mathcal{T}} = S_{\mathcal{T}}^*, S'_{\mathcal{T}}, \sigma_{\mathcal{T}}, \mathcal{R}_{\mathcal{T}}) \right] \text{(since } \nu \text{ is uniformly random in } \text{lrrel}(S_{\mathcal{T}}^*)) \\ &\leq \mathbf{E}_{S_{\mathcal{T}}^*} \left[ \frac{1}{|\text{lrrel}(S_{\mathcal{T}}^*)|} I(Z_{\mathcal{T}}^{\text{lrrel}} : \Pi | Z_{\mathcal{T}}^{\text{Special}}, S_{\mathcal{T}} = S_{\mathcal{T}}^*, S'_{\mathcal{T}}, \sigma_{\mathcal{T}}, \mathcal{R}_{\mathcal{T}}) \right] \\ &\quad \text{(by Observation IV.2)} \\ &\leq \max_{S_{\mathcal{T}}^*} \left( \frac{1}{|\text{lrrel}(S_{\mathcal{T}}^*)|} \right) \mathbf{E}_{S_{\mathcal{T}}^*} \left[ I(Z_{\mathcal{T}}^{\text{lrrel}} : \Pi | Z_{\mathcal{T}}^{\text{Special}}, S_{\mathcal{T}} = S_{\mathcal{T}}^*, S'_{\mathcal{T}}, \sigma_{\mathcal{T}}, \mathcal{R}_{\mathcal{T}}) \right] \\ &\leq \max_{S_{\mathcal{T}}^*} \left( \frac{1}{|\text{lrrel}(S_{\mathcal{T}}^*)|} \right) \times I(Z_{\mathcal{T}}^{\text{lrrel}} : \Pi | Z_{\mathcal{T}}^{\text{Special}}, S_{\mathcal{T}}, S'_{\mathcal{T}}, \sigma_{\mathcal{T}}, \mathcal{R}_{\mathcal{T}}) \\ &\leq \max_{S_{\mathcal{T}}} \left( \frac{1}{|\text{lrrel}(S_{\mathcal{T}})|} \right) \times \text{IRIC}(\Pi, \mathcal{T}) \end{aligned}$$

This finishes the proof of Theorem V.1.  $\blacksquare$

## VI. PUTTING IT TOGETHER

**Definition VI.1.** For an AND-OR tree  $\mathcal{T}$ , let  $R_\delta(\mathcal{T})$  denote the minimum communication complexity of a  $\delta$ -error randomized protocol computing  $\mathcal{T}(X, Y)$ .

**Lemma VI.2.** For a  $c$ -balanced tree  $\mathcal{T}$  with  $n$  leaves,  $\min_{S_{\mathcal{T}}} |\text{Irrel}(S_{\mathcal{T}})| \geq c^d \cdot n$

*Proof:* At each node  $v$ , the choice of the selector  $s_v$  forbids every leaf in the subtree  $T_{s_v}$  from belonging to  $\text{Irrel}(S_{\mathcal{T}})$ . However, if the node  $v$  is  $c$ -balanced, then irrespective of the choice of  $s_v$ , at least  $c$ -fraction of the leaves of  $v$  survive.

After the choice of the selector  $s_\rho$  at the root  $\rho$ , at least  $cn$  leaves survive. Now the choice of the selectors at depth 2 would prune away  $1 - c$ -fraction of the remaining leaves. Therefore at least  $c^2n$  leaves survive the choice of selectors at the depth 2. Extending the argument shows that for a depth  $d$  tree at least  $c^d \cdot n$  leaves belong to  $\text{Irrel}(S_{\mathcal{T}})$ .  $\blacksquare$

The following theorem follows easily from the above Lemma VI.2, Theorem V.1 and Theorem VII.1.

**Theorem VI.3.** For a  $c$ -balanced AND-OR tree  $\mathcal{T}$  of depth  $d$ ,  $R_\delta(\mathcal{T}) \geq \Omega(n(1 - 2\sqrt{\delta})^2(\frac{c}{4})^d)$

We now finish the proof of the main theorem, Theorem I.1, as follows. Given an arbitrary tree  $\mathcal{T}$  of depth at most  $d$ , we will perform a sequence of pruning operations to obtain a  $c$ -balanced tree  $\mathcal{T}'$  of depth at most  $d$  such that the communication complexity of  $\mathcal{T}$  is at least as much as that of  $\mathcal{T}'$ . This is shown in Claim VI.6 below. Combining the claim with Theorem VI.3 above yields Theorem I.1.

We begin by introducing the pruning operations.

**Observation VI.4.** (*Pruning Operation I*) Let  $\mathcal{T}$  be an AND-OR tree. Let  $\mathcal{T}' = \mathcal{T} - \mathcal{T}_v$  denote the AND-OR tree obtained by deleting the subtree rooted at  $v$  from  $\mathcal{T}$ . Then  $R_\delta(\mathcal{T}) \geq R_\delta(\mathcal{T}')$ .

**Observation VI.5.** (*Pruning Operation II*) If the root  $\rho$  of the tree  $\mathcal{T}$  has a single child  $v$ , then  $R_\delta(\mathcal{T}) \geq R_\delta(\mathcal{T}')$ .

We now state the claim that allows us to reduce communication lower bounds for general AND-OR trees to the case of  $\frac{1}{2}$  balanced trees (the proof is omitted).

**Claim VI.6.** Given an arbitrary AND-OR tree  $\mathcal{T}$  of depth at most  $d$  with  $n$  leaves, there exists a sequence of pruning operations such that the resulting tree  $\mathcal{T}'$  is a  $\frac{1}{2}$ -balanced AND-OR tree of depth at most  $d$  with at least  $\frac{n}{2^{2d}}$  leaves.

## VII. GADGET DISTRIBUTIONS

We begin by describing the distribution of inputs  $D_i$  to a complete *binary tree* of height  $i$  which served as a ‘‘gadget’’ in the input distribution that we define for arbitrary  $c$ -balanced AND-OR trees.

### A. Gadget Distributions

Let  $T$  be the complete binary tree of height  $i$  whose leaves are children of OR gates. We identify the leaves of  $T$  with  $[2^i]$ . We will think of  $T$  as defining a function from  $\{0, 1\}^{2^i} \times \{0, 1\}^{2^i}$  to  $\{0, 1\}$  defined by  $T(X, Y) := T(X \wedge Y)$ .

To each non-leaf node  $v$  of the AND-OR tree  $T$ , we associate a canonical input  $(\alpha_v, \beta_v) \in \{0, 1\}^{2^{\text{ht}(v)}} \times \{0, 1\}^{2^{\text{ht}(v)}}$  to the subtree  $T_v$  rooted at  $v$ , defined as follows:

- If  $v$  has height 1 (and hence is an OR gate), and let  $L = \{v_1, v_2\}$  (with  $v_1 < v_2$  as elements of  $[2^i]$ ) be the set of children of  $v$ . Then, inputs  $\alpha_v, \beta_v \in \{0, 1\}^L$  are given by  $\alpha_v = \beta_v = 10$  (here the input  $rs$  denotes input  $r$  to  $v_1$  and  $s$  to  $v_2$ ).
- Otherwise,  $\alpha_v = \alpha_{v_1}\alpha_{v_2}$  and  $\beta_v = \beta_{v_1}\beta_{v_2}$ .

We have the following simple properties:

- $T_v(\alpha_v, \beta_v) = T_v(\overline{\alpha_v}, \overline{\beta_v}) = 1$ .
- $T_v(\overline{\alpha_v}, \beta_v) = T_v(\alpha_v, \overline{\beta_v}) = 0$ .

Let  $v_0$  be the root of the tree  $T$ . We now define the distribution  $D_i$  over inputs to the tree  $T$ :

#### The distribution $D_i$ :

- If  $v_0$  is an AND gate, then  $D_i$  is the uniform distribution over the set  $\{(\alpha_{v_0}, \beta_{v_0}), (\overline{\alpha_{v_0}}, \overline{\beta_{v_0}})\}$ .
- If  $v_0$  is an OR gate, then  $D_i$  is the uniform distribution over the set  $\{(\overline{\alpha_{v_0}}, \beta_{v_0}), (\alpha_{v_0}, \overline{\beta_{v_0}})\}$ .

It is easy to check that the above construction satisfies Property III.3. Now we will lower bound a certain information cost for the distribution  $D_i$ .

### B. Bounding the Information Cost

Now we will lower bound the information complexity of a binary AND-OR tree of depth  $d$  for the specific construction of gadget distribution outlined earlier.

First we fix some notation related to binary trees. Let  $\mathcal{L}$  be the set of leaves of the tree. Let  $\mathcal{L}^X$  be the  $X$ -variables,  $\mathcal{L}^Y$  be the  $Y$ -variables (they are not part of the tree: hence  $|\mathcal{L}^X| = |\mathcal{L}^Y| = |\mathcal{L}| = 2^d$ ). For a node  $v$  of  $T$ , let  $\mathcal{L}_v$  be the set of leaves that are descendants of  $v$ . Let  $\mathcal{L}_{\overline{v}}$  be the set of leaves that are not descendants of  $v$ . Recall that for  $\alpha \in \{0, 1\}^{\mathcal{L}_v}$  and  $\gamma \in \{0, 1\}^{\mathcal{L}_{\overline{v}}}$ , we have

defined  $\alpha\gamma$  to be the element of  $\{0,1\}^{\mathcal{L}}$  “extending” the partial assignments  $\alpha$  and  $\gamma$ .

If  $\gamma$  is an assignment to some of the leaves of the tree and  $v$  is a node of the tree, then  $\gamma|_v$  denotes the restriction of  $\gamma$  to the variables coming from the subtree rooted at  $v$ . Similarly,  $\gamma|_{\bar{v}}$  denotes the restriction of  $\gamma$  to all the input variables not in the subtree rooted at  $v$ .

For a node  $v$  of height  $i$ , we will denote by  $D_v$  the distribution  $D_i$ , thought of as a distribution over inputs to the subtree rooted at  $v$ . For a node  $v$  of the tree, we define the set  $\text{DSupp}(v) \subseteq (\{0,1\}^{\mathcal{L}_v})^2$  be the set of those assignments  $(\gamma, \delta)$  to  $\mathcal{L}_v^X, \mathcal{L}_v^Y$ , such that for every ancestor  $u$  of  $v$ , letting  $w$  be the child of  $u$  that is not an ancestor of  $v$ , we have  $(\gamma|_w, \delta|_w) \in \text{support}(D_w)$ .

We can now prove the theorem lower bounding the information complexity of binary AND-OR trees.

**Theorem VII.1.** *Let  $\Pi$  be a  $\delta$ -error randomized communication protocol for the binary AND-OR tree of height  $d$ . Then  $\text{IRIC}(\Pi, T) \geq \frac{1}{8} \cdot (1 - 2\sqrt{\delta})^2 \cdot \frac{1}{4^d}$ .*

*Proof:* Recall

$$\begin{aligned} \text{IRIC}(\Pi^{\text{bin}}, T) &= I(Z_\nu : \Pi^{\text{bin}} | Z_{T-\{\nu\}}, S_T, \sigma_T, \mathcal{R}^{\text{bin}}) \\ &= \mathbf{E}_{i \in \{1,2\}} \mathbf{E}_{v \in \mathcal{L}} \mathbf{E}_{(\gamma, \delta) \in \text{DSupp}(v)} I((X_v, Y_v) : \Pi | S_T, \\ &\quad \nu = v, Z_{T-\{v\}} = (\gamma, \delta), \sigma_v = i, \mathcal{R}^{\text{bin}}), \end{aligned}$$

where the distribution of  $(\gamma, \delta)$  is uniform over  $\text{DSupp}(v)$ . First note that we may fix the value of  $\mathcal{R}^{\text{bin}}$  to some value without increasing the IRIC. Henceforth we assume that it has been fixed. Now, taking cases on  $i \in \{1, 2\}$ , we may write this expression as

$$\begin{aligned} &\frac{1}{2} \cdot \left( \mathbf{E}_{v \in \mathcal{L}} \mathbf{E}_{(\gamma, \delta) \in \text{DSupp}(v)} \left[ I((B, 0) : \Pi | S_T, \nu = v, \right. \right. \\ &\quad \left. \left. Z_{T-\{v\}} = (\gamma, \delta)) \right] \right. \\ &\quad \left. + \mathbf{E}_{v \in \mathcal{L}} \mathbf{E}_{(\gamma, \delta) \in \text{DSupp}(v)} \left[ I((0, B) : \Pi | S_T, \nu = v, \right. \right. \\ &\quad \left. \left. Z_{T-\{v\}} = (\gamma, \delta)) \right] \right), \end{aligned}$$

where  $B$  is a random bit.

For an input  $(x, y) \in \{0,1\}^{\mathcal{L}^X} \times \{0,1\}^{\mathcal{L}^Y}$ , we let  $P(x, y)$  be the probability distribution of the random variable  $\Pi(x, y)$ . For ease of notation we will denote joint assignments  $(\alpha, \beta, \dots)$  by  $\alpha\beta\dots$  when there is no confusion. Applying Lemma II.4, we may bound the first term in the parenthesis from below by

$$\mathbf{E}_{v \in \mathcal{L}} \mathbf{E}_{(\gamma, \delta) \in \text{DSupp}(v)} \left[ h^2(P(0\gamma, 0\delta), P(1\gamma, 0\delta)) \right]$$

and the second term is bounded below by

$$\mathbf{E}_{v \in \mathcal{L}} \mathbf{E}_{(\gamma, \delta) \in \text{DSupp}(v)} \left[ h^2(P(0\gamma, 0\delta), P(0\gamma, 1\delta)) \right].$$

Thus, combining these two expressions and applying the Cauchy-Schwarz inequality, we get

$$\begin{aligned} \text{IRIC}(\Pi^{\text{bin}}, T) &\geq \frac{1}{8} \mathbf{E}_{v \in \mathcal{L}} \left( \mathbf{E}_{(\gamma, \delta) \in \text{DSupp}(v)} \left[ \right. \right. \\ &\quad \left. \left. h(P(0\gamma, 0\delta), P(1\gamma, 0\delta)) + h(P(0\gamma, 0\delta), P(0\gamma, 1\delta)) \right] \right)^2. \end{aligned} \quad (8)$$

For any internal node  $v \in T$ , and  $(\gamma, \delta) \in \text{DSupp}(v)$ , define

$$\begin{aligned} \varepsilon(v, \gamma, \delta) &= \sum_{\substack{w \in \mathcal{L}_v \\ (\gamma', \delta') \in \text{DSupp}(w) \\ \gamma'|_{\bar{v}} = \gamma, \delta'|_{\bar{v}} = \delta}} \mathbf{E} \left[ \right. \\ &\quad \left. h(P(0\gamma', 0\delta'), P(1\gamma', 0\delta')) + h(P(0\gamma', 0\delta'), P(0\gamma', 1\delta')) \right]. \end{aligned}$$

By equation (8), letting  $\rho$  be the root of  $T$ ,

$$\text{IRIC}(\Pi^{\text{bin}}, T) \geq \frac{1}{8} \cdot \frac{1}{2^d} \cdot \varepsilon^2(\rho, \emptyset, \emptyset). \quad (9)$$

It remains to prove a lower bound on  $\varepsilon(\rho, \emptyset, \emptyset)$ . Such a lower bound will be derived from the following lemma.

**Lemma VII.2.** *For each internal node  $v \in T$ , for any  $(\gamma, \delta) \in \text{DSupp}(v)$ , the following four distributions are pairwise  $\varepsilon(v, \gamma, \delta) \cdot \sqrt{2}^{\text{ht}(v)}$  close in Hellinger distance:*

- $P(\alpha_v \gamma, \beta_v \delta)$ .
- $P(\alpha_v \gamma, \bar{\beta}_v \delta)$ .
- $P(\bar{\alpha}_v \gamma, \beta_v \delta)$ .
- $P(\bar{\alpha}_v \gamma, \bar{\beta}_v \delta)$ .

This lemma is proved by induction on  $\text{ht}(v)$ . The proof is omitted in this extended abstract.

Note that  $T_\rho(\alpha_\rho, \beta_\rho) \neq T_\rho(\alpha_\rho, \neg\beta_\rho)$ , and so Proposition II.3 implies that

$$h^2(P(\alpha_\rho, \beta_\rho), P(\alpha_\rho, \neg\beta_\rho)) \geq 1 - 2\sqrt{\delta}.$$

The previous Lemma implies that  $\sqrt{2}^d \varepsilon(\rho, \emptyset, \emptyset) \geq 1 - 2\sqrt{\delta}$ , and hence by Equation (9)

$$\text{IRIC}(\Pi, T) \geq \frac{1}{8} (1 - 2\sqrt{\delta}) \cdot \frac{1}{4^d}. \quad \blacksquare$$

## REFERENCES

- [1] B. Kalyanasundaram and G. Schnitger, “The probabilistic communication complexity of set intersection,” *SIAM Journal on Discrete Math*, vol. 5, no. 5, pp. 545–557, 1992.
- [2] A. A. Razborov, “On the distributional complexity of disjointness,” *Theoretical Computer Science*, vol. 106, no. 2, pp. 385–390, 1992.

- [3] T. S. Jayram, R. Kumar, and D. Sivakumar, “Two applications of information complexity,” in *STOC*. ACM, 2003, pp. 673–682.
- [4] A. Chakrabarti, Y. Shi, A. Wirth, and A. C.-C. Yao, “Informational complexity and the direct sum problem for simultaneous message complexity,” in *Proc. 42nd IEEE Annual Symposium on Foundations of Computer Science*, 2001, pp. 270–278.
- [5] Z. Bar-Yossef, T. S. Jayram, R. Kumar, and D. Sivakumar, “An information statistics approach to data stream and communication complexity,” *J. Comput. Syst. Sci.*, vol. 68, no. 4, pp. 702–732, 2004.
- [6] E. Kushilevitz and N. Nisan, *Communication Complexity*. Cambridge University Press, 1997.
- [7] N. Alon, Y. Matias, and M. Szegedy, “The space complexity of approximating the frequency moments,” *Journal of Computer and System Sciences*, vol. 58, no. 1, pp. 137–147, 1999.
- [8] D. P. Woodruff, “Optimal space lower bounds for all frequency moments,” in *SODA*, J. I. Munro, Ed. SIAM, 2004, pp. 167–175.
- [9] T. Lee and A. Shraibman, “Disjointness is hard in the multi-party number-on-the-forehead model,” in *IEEE Conference on Computational Complexity*. IEEE Computer Society, 2008, pp. 81–91.
- [10] A. Chattopadhyay and A. Ada, “Multipart communication complexity of disjointness,” *CoRR*, vol. abs/0801.3624, 2008.
- [11] M. David, T. Pitassi, and E. Viola, “Improved separations between nondeterministic and randomized multipart communication,” in *APPROX-RANDOM*, vol. 5171, 2008, pp. 371–384.
- [12] P. Beame and D.-T. Huynh-Ngoc, “Multipart communication complexity of  $AC^0$ ,” Electronic Colloquium on Computational Complexity, Tech. Rep., 2008. [Online]. Available: <http://eccc.hpi-web.de/eccc-reports/2008/TR08-061/index.html>
- [13] A. A. Sherstov, “Separating  $AC^0$  from depth-2 majority circuits,” in *STOC*, 2007, pp. 294–301.
- [14] —, “The pattern matrix method for lower bounds on quantum communication,” in *STOC*, R. E. Ladner and C. Dwork, Eds. ACM, 2008, pp. 85–94.
- [15] A. V. Aho, J. D. Ullman, and M. Yannakakis, “On notions of information transfer in vlsi circuits,” in *Proc. 15th Annual ACM Symposium on the Theory of Computing*, 1983, pp. 133–139.
- [16] H. Buhrman and R. de Wolf, “Complexity measures and decision tree complexity: A survey,” 2000, manuscript, to appear in *Theoretical Computer Science*.
- [17] M. Snir, “Lower bounds for probabilistic linear decision trees,” *Theoretical Computer Science*, vol. 38, pp. 69–82, 1985.
- [18] M. Saks and A. Wigderson, “Probabilistic Boolean decision trees and the complexity of evaluating game trees,” in *Proc. 27th IEEE Symposium on Foundations of Computer Science*, 1986, pp. 29–38.
- [19] R. Heiman and A. Wigderson, “Randomized vs. deterministic decision tree complexity for read-once Boolean functions,” in *Proc. 6th Structure in Complexity Theory Conference*, 1991, pp. 172–179.
- [20] R. Heiman, I. Newman, and A. Wigderson, “On read-once threshold formulae and their randomized decision tree complexity,” in *Proc. 5th Structure in Complexity Theory Conference*, 1990, pp. 78–87.
- [21] N. Leonardos and M. Saks, “Private communication,” 2009.
- [22] F. Ablayev, “Lower bounds for one-way probabilistic communication complexity and their application to space complexity,” *Theoretical Computer Science*, vol. 157, no. 2, pp. 139–159, 1996.
- [23] M. Saks and X. Sun, “Space lower bounds for distance approximation in the data stream model,” in *Proc. of the 34th Annual ACM Symposium on Theory of Computing*, 2002, pp. 360–369.
- [24] R. Bar-Yehuda, B. Chor, E. Kushilevitz, and A. Orlitsky, “Privacy, additional information, and communication,” *IEEE Transactions on Information Theory*, vol. 39, no. 6, pp. 1930–1943, 1993.